

Verification and Validation Methodology of Real-time Adaptive Neural Networks for Aerospace Applications

P. Gupta, Ph.D¹, K.A. Loparo, Ph.D², D. Mackall^{3*}, J. Schumann, Ph.D⁴, F. R. Soares, Ph.D⁵

¹QSS, Inc./NASA AMES Research Centre, Mail Stop 269-3, Moffett Field, CA 94035, U.S.A

E-mail: pgupta@email.arc.nasa.gov

²Professor, Case Western Reserve University. (CWRU), EECS Department, Cleveland, Ohio 44106-7071, U.S.A

E-mail: kal4@cwru.edu

^{3*}Retired in December 2003, NASA Dryden Flight Research Center, P. O. Box 273, Edwards CA 93523-0273, U.S.A

⁴USRA-RIACS/NASA AMES Research Centre, Mail Stop 269-3, Moffett Field, CA 94035, U.S.A

E-mail: schumann@email.arc.nasa.gov

⁵Contek Research, Inc., P. O. Box 88758, Los Angeles, CA 90009, U.S.A

E-mail: foia@contekresearch.com

Abstract

Recent research has shown that adaptive neural based control systems are very effective in restoring stability and control of an aircraft in the presence of damage or failures. The application of an adaptive neural network with a flight critical control system requires a thorough and proven process to ensure safe and proper flight operation. Unique testing and performance evaluation tools have been developed as part of a process to perform verification and validation of real time adaptive neural networks. The tools will help in FAA certification and in the successful deployment of neural network based adaptive controllers in safety-critical applications. The process to perform verification and validation process is evaluated on a typical neural adaptive controller, and the results are compared.

1 Introduction

Several of the solutions developed during the F-8 digital fly-by-wire program in the 1970s and 80s can be directly applied to the current research in neural flight control. A brief historical review of the digital fly-by-wire program provides a starting point for developing a verification and validation approach for neural flight control.

The primary challenges addressed by the digital flight-by-wire program included:

- Hardware reliability
- Software reliability
- System validation

The solutions to these challenges were based on the lessons learned from a major previous program, the Apollo program. The first phase of the digital fly-by-wire program actually used a single Apollo flight computer and sensor set. Software reliability was insured by using the software development processes from Apollo. Likewise System validation was accomplished in a similar manner as the Apollo program.

The second phase of the digital fly-by-wire program addressed hardware reliability in a new manner: it used three identical flight control computers to provide reliability. The new triple redundant hardware approach introduced a new software component to handle detection, isolation, and reconfiguration due to hardware failures. With the new computers, based on the

Space Shuttle's flight computers, the software development process evolved from the baseline approach used during the Apollo program [1, 2, 3]

Safety-critical applications require that the system can cope with unforeseen catastrophic changes or slow degradation over the time. Although aircraft flight control design by classical techniques has produced reliable and effective control systems, the desire to create new concepts in aircraft design has resulted in increased research in advanced control techniques. The new concepts require the aircraft to perform over an increased range of operating conditions with variations in dynamic pressure and nonlinear aerodynamic phenomena, and the use of nonlinear actuation systems that increase the complexity of the control design. Over the years, a variety of approaches to control such processes have been developed (e.g., adaptive control). Adaptive control is concerned with the construction of robust controllers for processes, which change over time. All changes in the system dynamics are processed by the controller with the aim that the desired response to the control input is kept. In the recent past, neural network based controllers are being used for such a purpose. Adaptive neural network techniques have demonstrated the potential as a good candidate for controlling nonlinear and complex aircraft flight system; thus many research centers and universities have been investigating and experimenting with these techniques and their applications [4,5,6,14]. It is often seen that neural networks are proposed as a tool for adaptive control of nonlinear systems with time-varying dynamics. A major benefit of such controllers is an ability to adapt to unforeseen events, e.g., stuck rudder in case of aircraft. Despite the advantages of on-line trained neural network based systems, the lack of methods to perform certification, verification and validation of such systems severely limits their use in real-life.

Intelligent adaptive flight control is faced with many of the same challenges with the introduction of neural networks as a part of control system. The neural networks must reside in redundant flight hardware, and must operate correctly in the presence of failures or unforeseen circumstances. The implementation of the neural network into software must be reliable by using a proven software development process that has evolved over the years. Furthermore, the unique adaptive nature of neural networks must be verified. Finally, validation of the entire system, including the adaptive neural network, the flight control system, and the dynamics of the aircraft under nominal and failure conditions must be proven to be safe, if not safer than their conventional counterparts.

The technology being developed using neural flight control must be based on proven processes and demonstrated in flight test programs. Then the overall benefits of neural flight control will be found in tomorrow's aircraft, just as the benefits of digital fly-by-wire are found in aircraft today.

2. Verification & Validation of Neural Network

Before artificial neural networks can be used in safety-critical systems a verification and validation process must be developed and later incorporated into the certification process. Present day controllers for safety critical applications, such as aircraft control, undergo extensive testing to qualify their operation. These systems must satisfy rigorous V&V requirements. Due to the special nature of on-line trained neural networks, traditional V&V techniques are not sufficient to provide guarantees for desired behavior. Therefore there is a need to develop new V & V techniques for adaptive neural systems.

The approach to verification and validation of neural network control system is based on the knowledge of qualifying previous control systems and unique characteristics of neural networks. The approach is also dependent on how the neural networks affect the performance of the control system. There are three main considerations in determining the approach to verification and validation of neural networks. At the highest level, is the overall architecture of the neural network within the flight control system and its performance. The second consideration is the type of neural network being used. The last consideration is probably the most important, it is the method used to adapt the neural network to solve the problem at hand. This is normally called the learning algorithm.

2.1 Issues involved in V & V of Neural Network

Testing to ensure that the neural network always learns correctly is therefore critical. The adaptation law adjusts the neural network weights to solve a particular task, or in a flight control system to minimize a particular error. The learning algorithm used in real-time adaptive flight control systems is derived from nonlinear analysis of the flight control system and the neural network. The first step in verifying a real-time adaptive flight control system is to verify the learning algorithm by using sensitivity and noise analysis.

Once the learning algorithm has been verified, the second step is to evaluate the neural network architecture with its learning algorithm. Testing of the neural network architecture utilizes techniques such as sensitivity and noise analysis.

Validating the aircraft systems as a whole, which uses a real-time adaptive neural network to correct for surface failures or aircraft damage, requires an extensive test matrix. Stability and control of the aircraft must be evaluated across the entire flight envelope, and include all possible surface failures, as well as possible damage consequences. The damage matrix can become quite complex, including such items as: loss of lift, pitch, roll, and yaw moments, asymmetrical drag, and failures in thrust. Validating such a large test matrix requires automating the tasks and performing test using nonlinear simulations, as well as tests on actual flight article. Although these approaches are very promising with respect to their performance and their capability to adapt, the question how to ensure the correct behavior of such a system has not been addressed in detail. In general, each piece of software, which is used in safety-critical applications, has to go through a rigorous certification process to make sure that the certification authority is convinced that the system with such software is safe and ensure that the software cannot fail. The behavior of adaptive systems cannot be simply characterized. These systems must satisfy a stringent V & V, which assigns bounds to their output error under all operating conditions and guarantees that no combination of inputs will result in undesirable/catastrophic output.

Current V&V methods, which rely heavily on testing, make up a large fraction of development costs for modern aircraft, yet they do not strictly guarantee performance. The main challenge in the design of V&V methods for adaptive control system is due to the fact that such systems are nonlinear, and as a result, all aspects of the control design including the delineation of the performance specifications, the controller structure, the development of stability measures, and the guarantees in performance under different operating conditions are quite difficult.

Due to the special nature of an on-line trained adaptive neural network, traditional V&V techniques are not sufficient to provide guarantees for the safe and desired behavior. Therefore, there is need to develop V&V techniques for such systems [4,7,8,11, 13, 14].

3. Development of the Tool for the neural network-based flight control system

3.1 A Generic NN Analysis Tool

A NN analysis tool was built using Matlab/Simulink and made part of the main simulation program. Development of a Matlab/Simulink based tool facilitated analysis and performance evaluation of the of the a adaptive/NN flight controller. The NN analysis tool implemented in Matlab/Simulink, includes the options of plotting the trajectory of the NN weights, the NN estimation of the inversion error, the approximation error, time history of the NN weights and aircraft parameters the trajectory of system states p, q and r. (roll, pitch and yaw rates). We have defined the basic features of the tool with the objective to provide access to the important parameters of the NN and adaptation algorithm. Furthermore, various measures of performance are defined and will be included as part of the tool.

A tool includes time history plots of the weights and cross plotting of variables that shows the boundedness of the weights and the tracking error for the three channels, in addition to the NN estimation for the inversion errors. It includes the capability to input (gains) perturbations for sensitivity analysis i.e. Gains and Noise Analysis in the “NN tool analysis”. The Noise analysis will introduce a White Noise to the gains in each channel. This demonstrates how much the cancellation of the inversion errors changed with the addition of the white noise.

3.2 Confidence tool

The Confidence tool was developed to measure the performance of neural network during operation. The primary motivation of developing this tool was to enable use of neural networks in applications such as flight control software. The tool is a part of our layered approach [11] for V & V of neural network based adaptive systems. Our tool dynamically calculates the current performance characteristics of the system and provides a dynamic measure of how reliable the current approximation of the system is. The dynamic measurement checks all inputs and outputs values of the neural network and determines if the result of the neural network is reliable. Our tool is based on the statistical model of the output of the neural network.

The process involves in calculating the confidence interval around the output of the neural network. Confidence intervals provide a way to quantify confidence in the estimate of the neural network output. For a desired degree of confidence, (namely, for a given probability), a confidence interval is a prediction of the range of the output of a model where the actual value exists. With an assumption of a normal distribution of the errors, confidence intervals can be calculated for neural networks. Calculation of the confidence intervals can include the effects of developing neural network model from noisy data. We consider the application of Bayesian techniques for estimation of a dynamic measure on the reliability of the neural network’s current approximation [11,12]. The width of the error bars gives the measure about the performance of the neural network. The width of the error bars depends on the locality density of input data, with the error bars increasing in magnitude in regions of the input space having low density.

4.0 Simulation

The Intelligent Flight Control System (IFCS) system uses a direct adaptive neural network based approach to flight control. The main goal is to develop and flight demonstrate adaptive neural network based flight control system. This approach incorporates neural network that uses flight control system feedback errors to provide adjustments to improve aircraft performance in both normal flight and with system failures. The flight project results will be utilized in an overall strategy aimed at advancing neural network flight control technology to new aerospace system designs including civil and military aircraft, reusable launch vehicles, uninhabited vehicles, and space vehicles.

The concept is based on a dynamic inversion controller with a model-following command path. The feedback errors are regulated with a proportional plus integral (PI) controller. This basic system is augmented with an adaptive neural network that operates directly on the feedback errors. The adaptive neural network adjusts the system for miss-predicted behaviour, or changes in behaviour resulting from damage [10, 12].

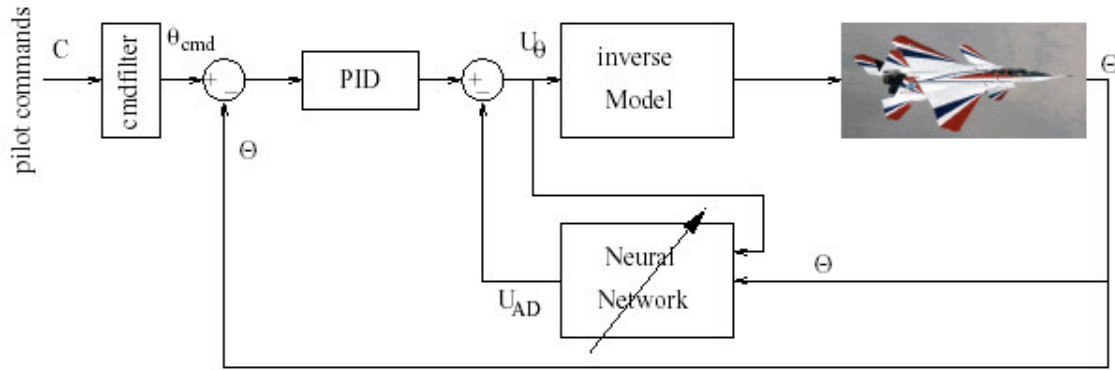


Figure 1: Simplified IFCS control architecture

Figure 1 shows a block diagram of a simplified version of the controller [7]. This controller is based upon a dynamic inverse model. The commands C issued by the pilot are pre-filtered, θ_{cmd} and, in combination with the aircraft sensor readings Θ , a desired command output U_θ is calculated. This value is then fed into the inverse model of the aircraft (realized using a linear mapping or a pre-trained neural net). This inverse model then provides the actual command values, which are fed to the aircraft's control surfaces. In the nominal and idealized case, the inverse model is an exact inverse function of the aircraft dynamics. Due to modeling inaccuracies or changes in the aircraft dynamics, there are deviations. In this case, the neural network kicks in: attempting to learn from the sensor readings Θ and the previous command output U_θ how to minimize the deviation. Thus, the neural network's output U_{AD} is used to adjust the value of U_θ .

4.1 The Simulation Result

Results of the simulation using the developed tools are shown in the following figures. Two cases: Case 1: No failure or nominal case; Case 2: Failure introduced as Differential Stabilator at 8.5 degrees. In the nominal case, the neural network produces some output to accommodate for modeling inaccuracies shown in Figs. 2a-2c. During that time of adaptation, the confidence decreases. Figures (4a-4c) depict scenarios when a failure occurs (a control surface got stuck at angle 8.5 degrees). In this case, the adaptive neural network output needs to be substantially large to counteract the damage. Also, the width of the error band is much larger, indicating a low confidence.

5.0 Conclusions

In this paper, we have reported ongoing work on the development of tools for the verification and validation of neural network based adaptive controllers for aerospace applications. It is our belief that these tools will be able to help in V & V of neural network based controller and help in the successful deployment of aerospace systems in safety critical applications. We have developed the tools, which can monitor the neural networks' behavior dynamically and calculate a confidence interval, indicating the reliability and sensitivity to flight control parameters. Our tools provide assistance to establish reliability of a neural network based adaptive control system and will enhance the understanding of performance of the adaptive systems, addressing the detection and possible prevention of catastrophic failures. These tools will attempt not only to guarantee the performance of the adaptive systems at all locations of the flight domain but also enable the design of suitable V & V test techniques of these flight systems. These tools have been integrated into an Automated Neural Flight Control System Test Tool (ANCT) to help test engineers validate neural flight controllers in various flight conditions, quantify performance, and determine regions of stability.

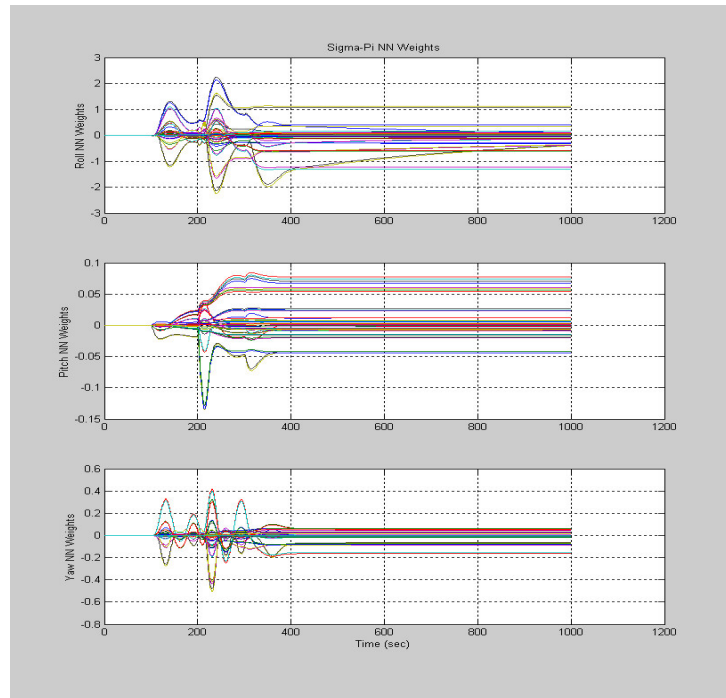


Figure 1: Sigma Pi NN weights (no failure)

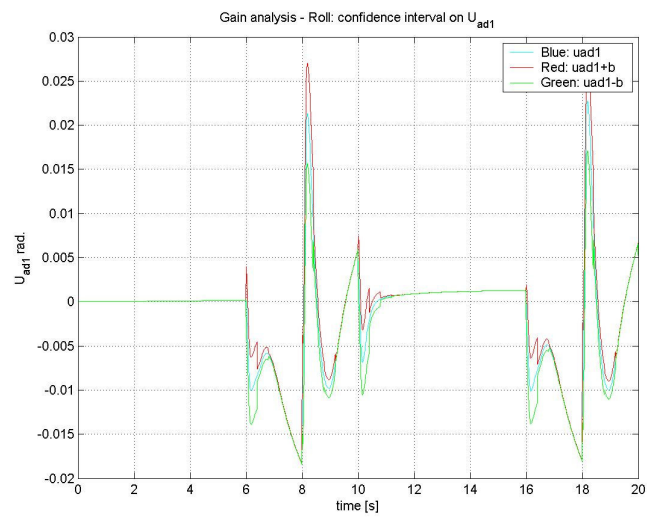


Figure 2A: Roll confidence interval (no failure)

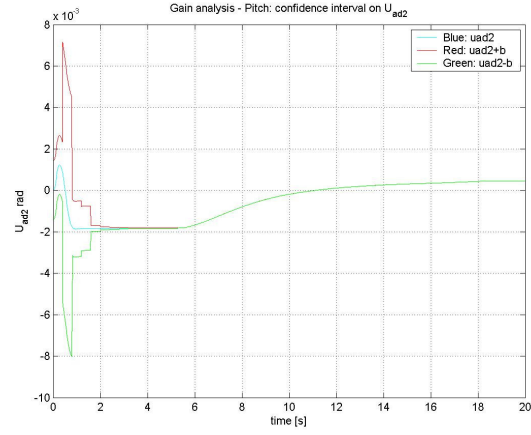


Figure 2B: Pitch confidence interval (no failure)

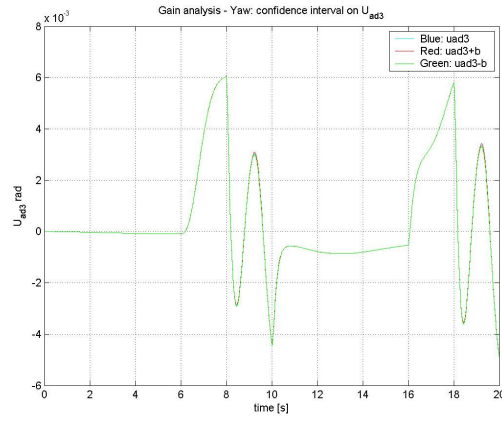


Figure 2C: Yaw confidence interval on (no-fail)

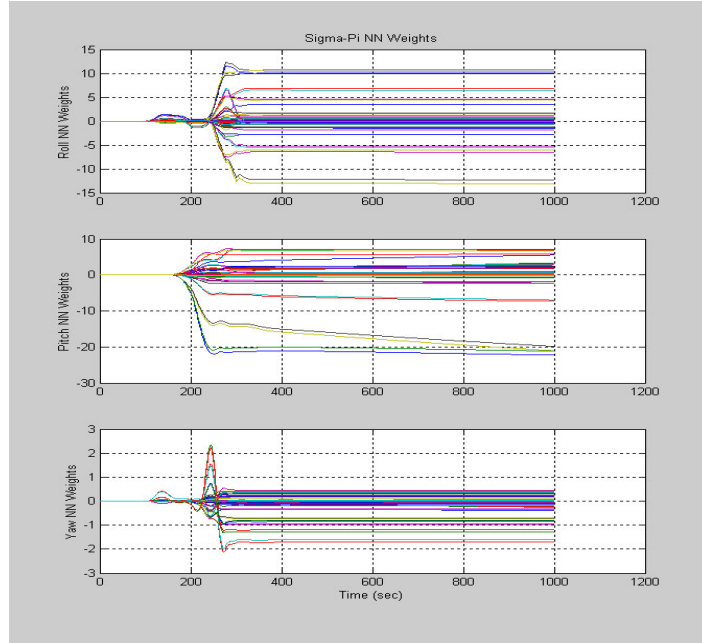


Figure 3: Sigma Pi NN weights (surface failure)

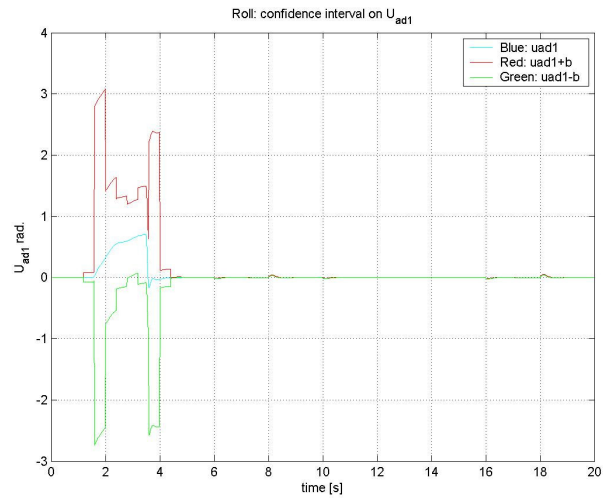


Figure 4A: Roll confidence interval (surface failure)

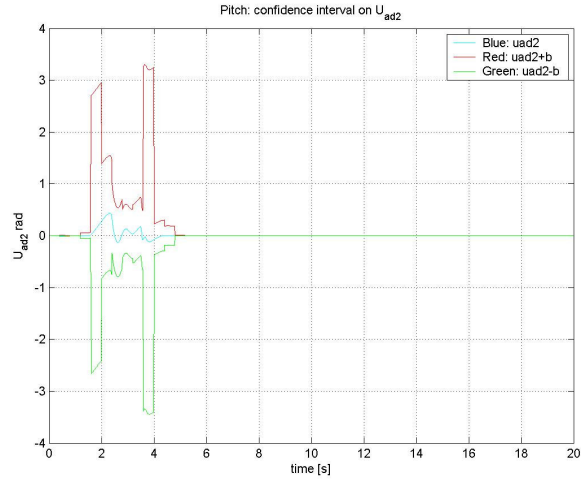


Figure 4B: Pitch confidence interval (surface failure)

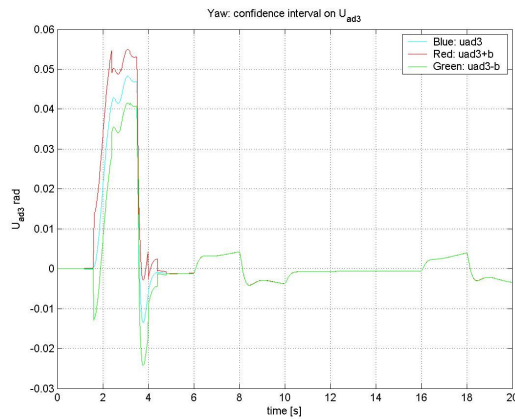


Figure 4C: Yaw confidence interval (surface failure)

Simulation Conditions

Fail Surface: Right Stabilator

Fail Position: -8.5 deg.

Start time of failure: 1.5 sec.

Stop time of failure: 3.5 sec.

$M = .75/20k$ ft

Stab trim = +2.85 deg.

Canard trim = -3.94 deg.

$U_{ad} = uad$ (1=roll; 2=pitch; 3=yaw – adaptation error (NN output)

b – Confidence level “sigma”

References

- [1] RTCA DO-178B – *Overview of Aircraft and Engine Certification* p. 45
- [2] NASA Procedures and Guidelines NPG: 2820.DRAFT, *NASA Software Guidelines and Requirements as of 3/19/01* (Responsible Office: Code AE/Office of the Chief Engineer), NASA Ames Research Center, Moffett Field, California, USA
- [3] *Software Considerations in Airborne Systems and Equipment Certification, Document No RTCA* (Requirements and Technical Concepts for Aviation) /DO-178B, December 1, 1992. (Copies of this document may be obtained from RTCA, Inc., 1140 Connecticut Avenue, Northwest, Suite 1020, Washington, DC 20036-4001 USA. Phone: (202) 833-9339)
- [4] R. Ben Ayed, A. Mili, G. Jiang, B. Cukic, Y. Liu, "Verifying Properties of On-Line Adaptive Systems," *Proceedings of ACS/IEEE International Conference on Computer Systems and Applications*, Tunis, July 2003.
- [5] C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon-Press, Oxford, 1995.
- [6] John Kanesisghe, John Bull, and Joseph Totah, *Generic Neural Flight Control and Autopilot System*, AIAA-2000-4281, Auguts 2000.
- [7] Y. Liu, B. Cukic, T. Menzies, S. Gururajan, M. Napolitano, "Validating an On-Line Adaptive System Using Support Vector Data Description," 15th International Conference on Tools with Artificial Intelligence, Sacramento, CA, November 2003.
- [8] Y. Liu, T. Menzies, B. Cukic, "Data Sniffing: Monitoring of Machine Learning for On-Line Adaptive Systems," 14th International Conference on Tools with Artificial Intelligence, Washington, DC, November 2002.
- [9] M. Norgaard, C. Jorgensen and D. J. Ross, *Neural Network Prediction of New Aircraft Design Coefficient*, NASA TM-112197, 1997.
- [10] R. Rysdyk and A. Calise, *Fault Tolerant Flight Control Via Adaptive Neural Network Augmentation*, American Institute of Aeronautics and Astronautics, AIAA-98-4483:1722-1728, 1998.
- [11] Johann Schumann, Pramod Gupta, Stacy Nelson. *On Verification and Validation of Neural Network Based Controllers*, Proc. of International Conf on Engineering Applications of Neural Networks, EANN '03 in Spain, pp. 40¹-47
- [12] L. H. Ungar, R. D. D. Veaux, and E. Rosengarten. *Estimating Prediction Intervals for Artificial Neural Networks*, Proc. of the 9th Yale Workshop on Adaptive and Learning Systems, 1996.

[13] W. Wen, J. Callahan, and M. Napolitano, *Towards Developing Verifiable Neural Network Controllers*, Proc. of the workshop on AI for Aeronautics and Space, 1996.

[14] C. Jorgensen. *Direct Adaptive Aircraft Control Using Neural Networks*. Technical Report TM-47136, NASA, 1997.

Acknowledgements

U.S. Government under contracts and/or grants by NASA Dryden Flight Research Center, Edwards, CA., U.S.A.

Symbols Definitions